WAIS Interface Protocol
Prototype Functional Specification

Version 1.5
April 23, 1990

Franklin Davis, Brewster Kahle, Harry Morris, Jim Salem, Tracy Shen
Thinking Machines Corporation

Rod Wang, John Sui, Mark Grinbaum
Dow Jones & Company, Inc.

Contents

1. Overview

The purpose of this interface is to establish an application level
(ISO 2) protocol for query/retrieval applications. The initial
implementation will provide a protocol for the DowQuest database
service provided by Dow Jones News Retrieval. Workstation interfaces
will be implemented on the Macintosh as part of the WAIS project (Wide
Area Information Server). The intention is to provide a sophisticated
and expandable computer-to-computer interface for future databases.

This protocol is based on the Z39.50-1988 ("the standard") Information
Retrieval Service Definitions and Protocol Specification for Library
Applications. Each section of this document includes references in
square brackets "[]" to the appropriate section(s) in the Z39.50
specification.

The standard specifies an Opens Systems Interconnection application
layer service definition and protocol specification for Information
Retrieval. The Information Retrieval protocol allows an application
on one computer to query the database of another computer. The
protocol specifies the procedures and structures for the intersystem
submission of a search request (including the syntax of the query),
request for the transmission of database records located by a search,
the responses to the request, access control, and resource control.

This is the last version of the WAIS protocol to be based on the
Z39.50 standard. The next version will implement the newer SR-1
standard, which is based on Z39.50, but is written in ASN.1.

The WAIS extensions to the standard are primarily to support
"relevance feedback" queries. (The standard currently supports a
boolean query syntax.) The Present facility is not used, in order to
allow the target system to be "stateless" (to always delete Result-
Sets.) Instead, a Type-1 query is used for text retrieval. In order
to retrieve document number xxx, a search is performed with a query
specifying that System-Control-Number=xxx.

The WAIS extensions also enable the origin to request a range of
document text. The Type-1 query is used as described in the previous
paragraph with the addition of Chunk-Code parameters. The portion of
the document that matches the Chunk-Code values will be returned, e.g.
"System-Control-Number=xxx AND Line>1000 AND Line <= 2000" would
return lines 1001 through 2000 of document xxx.

This protocol requires the target system to return unique document IDs
in a Search-Response, labeled as System-Control-Number (see Appendix C
of the standard). These document IDs are used by the origin (user
interface) to specify documents when requesting display of a document
or in relevance feedback searches.

Retrieval of large documents dependsw on the ability to specify a
range of a document in a search. This will be specified with an
extension called "Chunks." This version of the protocol does not have
a method for the origin and target to negotiate the available chunk
types. Three chunk types are currently defined for DowQuest: Byte,
Line, and Paragraph.

For efficiency reasons it is useful to refer to a document range with
large "chunks" that have been marked in the text by the target system.
The chunk markers and IDs are not displayed to the user, but are used
by the origin when the user selects a range of a document for a
relevance feedback query. The Init-Response APDU is extended to
provide "chunk" markers and sizes which may be used to specify
document ranges in relevance feedback queries.

The User Information part of APDUs is used in more complex ways in
this extension than was originally envisioned in the standard. In the
standard, the User Information part was a single Element of type
"any." The WAIS protocol extensions uses User-Information-Field
preceding the set of elements in the user information part of an APDU.
This is the length in bytes of all the following elements, excluding
the User-Information-Length element.

1.1 Supported Facilities

For the June 1990 target delivery date of the prototype WAIS system, DowQuest will support only 2 facilities from the Z39.50 specification.

The "Initialization Facility" [3.2.1] includes an "Init APDU" [4.1.1.1, table A2] and an "Init-Response APDU" [4.1.1.2, Table A3].

The "Search Facility" [3.2.2] includes a "Search APDU" [4.1.1.3, table A4] and a "Search-Response APDU" [4.1.1.4, table A5].

"APDU" means "Application Protocol Data Unit," which is a unit of data passed between an origin (user workstation) and target (database server). These and other terms are defined in section 2 of the Z39.50 specification.

The Search APDU will be extended to have a new query type: Type-3, "Relevance Feedback Query."

The Search-Response APDU will be modified to include new elements in Database-Records, including Document-IDs (used for relevance feedback) and other fields, specified in section 4 of this document.

1.2 Unsupported Facilities

The remaining 5 facilities from Z39.50 are not supported in the WAIS prototype.

The "Retrieval Facility" will not be supported in the Wais prototype. Document text will be retrieved using a Type-1 query based on System-Control-Number (document ID).

The "Result-Set-Delete Facility" is not needed because DowQuest will always delete all Result-Sets after returning a Search-Response APDU.

The "Access Control Facility" will not be supported. All users will have access to all data in DowQuest.

The "Accounting/Resource Control Facility" will not be supported. DowQuest responses have a maximum size.

The "Termination Facility" is not needed because DowQuest will not store any state about user sessions. Each request and response will be a complete transaction, independent of all others. Either the origin or the target may abort a session at any time.

1.3 Conformance with Version 1 of Z39.50

1.3.1 Extensibility

As specified in section 4.3 of the standard, WAIS systems will ignore unknown data elements and options in received Init APDUs.

1.3.2 Static Requirements

The DowQuest system will conform to the Static Requirements specified in section 4.4.1 of the standard, with extensions noted in this document, except that it will NOT support general boolean Type-1 queries. The Type-1 query will be used only for retrieval of documents based on System-Control-Number and Chunks.

1.3.3 Dynamic Requirements

WAIS systems will conform to the Dynamic Requirements specified in section 4.4.2 of the standard. There are restrictions on the Type-1 Query.

1.3.4 Statement Requirements

DowQuest will be capable of acting in the role of target. It supports

version 1 of the standard.

See section 1.2 of this document for unsupported facilities.

Result-Sets will always be unilaterally deleted by DowQuest. It will not accept Search APDUs specifying named result sets. Each input and response message pair is a complete, independent transaction. Thus, multiple users may share a single session, although the order of responses is not guaranteed to be the same order as the requests. If multiple users share a connection, the origin must use Reference-IDs to identify input/response message pairs.

DowQuest supports element set names in Search APDUs as specified in section 4 of this document.

The maximum number of database names that may be specified in a Search APDU will be determined by the implementors.

1.4 Errors in the Standard

Table A7 on p. 43 of the standard is a copy of table A6. Table A7 should contain the fields defined in 4.1.1.6, p. 23. Earlier versions of the WAIS protocol specification contained the same error in table B.6.

2. Initialization Facility

DowQuest will accept an Init APDU at any time, and will always respond with an Init-Response APDU. Since DowQuest is stateless, the Initialization facility is not required to begin a user session, but it may be used anytime to get the system parameters.

The Init-Response APDU may specify "chunk" parameters that may be used to specify a range of a document in a relevance feedback Type-3 Query. [??? The chunk negotiation needs to be defined more completely.]

The Init-Response APDU may also specify newline characters, non-displayable field markers, and highlight/non-highlight markers, and fields describing how often the target is updated and when the target is updated.

2.1 Init APDU

The Init APDU requests information about the database service [3.2.1, 4.1.1.1, and Table A2]. Since DowQuest is stateless, Init is not required to begin a user session.

The Options field must always have 0="will not use" for the Delete facility.

See Appendix B.1 of this document for an example Init APDU.

2.2 Init-Response APDU

The Init-Response APDU provides information about the database service [3.2.1, 4.1.1.2, and Table A3].

The Options field will always have 0="will not support" for the access-control and resource-control facilities.

Implementation-Name will be "DowQuest", and the Implementation-Version will be set by the implementors, to be updated as new versions are released.

Preferred-Message-Size and Maximum-Record-Size will be determined during the implementation.

See Appendix B.2 of this document for an example Init-Response APDU.

2.2.1 Chunk IDs

The User-Information-Field of the Init-Response APDU will contain four elements indicating ways the origin may specify a region of a document to be used in a relevance feedback Type-3 query. The region is composed of a range of "chunks" such as bytes or paragraphs. The elements are:

Search-Chunk-Code-Bitmap	O	bitmap
Present-Chunk-Code-Bitmap	[???] O	bitmap
Chunk-ID-Length	C	integer
Chunk-Marker	C	ASCII

Search-Chunk-Code-Bitmap specifies the chunk codes the target will accept in Type-1 Queries in Search APDUS requesting display of document regions. The bitmap indicates with a "1" in a bit position that the corresponding code number will be accepted by the target system. For example, to indicate that the target accepts accepts Chunk-Codes 1 and 3 in a Search APDU it would return Search-Chunk-Code-Bitmap with bits 1 and three set to 1 and all other bits 0.

Initially, four Chunk-Codes are defined. The default is 1 "Byte" (see section 5 of this document):

        Chunk-Code=0 "Document"

```
Chunk-Code=1 "Byte"
Chunk-Code=2 "Line"
Chunk-Code=3 "Paragraph"
```

(In the future this may be extended to include other measures, such as
Word, Page, or Chapter-1D. Other media such as audio might use chunks
such as Song-ID or Seconds. Video might use Frame or Scene-ID.)

Chunk-Code=1 "Byte" is the most general case. With this chunk size,
Chunk-Marker and Chunk-ID-Length are not used. The origin may
indicate ranges of a document in bytes by setting Chunk-Code=1 and
providing pairs of byte-offsets in a relevance feedback Type-3 query.
If any Chunk-Code > 1 is accepted, the target must also provide
Chunk-ID-Length and Chunk-Marker.

DowQuest will provide Chunk-Code=3 (Paragraph-ID) for relevance
feedback Type-3 Queries, and Chunk-Code=2 (Line) for text retrieval
Type-1 Queries.

[??? Need more general chunk mechanism for both tagged and counted
types, e.g. paragraphs are tagged, but lines are counted (each line is
"tagged" only by the presence of a newline). This will be addressed
in the next version of the protocol.]

2.2.2 Other Markers

DowQuest will also provide elements in the User-Information field of
the Init-Response APDU indicating various non-displayable marker
fields.   These include:

Highlight-Marker	O	ASCII
De-Highlight-Marker	C	ASCII
Newline-Characters	O	ASCII

If Highlight-Marker is present, De-Highlight-Marker is required.

2.2.3 Other Information Elements

WAIS targets may provide elements describing how often and when the
database is updated:

Update-Frequency	O	[???]
Update-Times	O	[???]
[??? pricing info?]	O	[???]

[The format and tags of these fields is TBD.]

3. Search Facility

3.1 Search APDU

The Search APDU will be implemented as defined in the standard [3.2.2, 4.1.1.3, and Table A4]. However, the Result-Set will always be deleted by DowQuest immediately after returning a Search-Response APDU, so the Replace-Indicator field in the Search APDU should be "on," an and Result-Set-Names is not used. Search APDUs may not refer to a Result-Set. This enables DowQuest to be stateless.

The Type-3 Relevance Feedback Query syntax is outside the scope of the standard. The syntax used by DowQuest is given in Appendix A.

DowQuest will support the Type-1 Query syntax, but not for general boolean queries. Only searches specifying System-Control-Number (and possibly Chunk ranges) are supported.

See Appendix B.3 of this document for an example Search APDU.

3.2 Search-Response APDU

The Search-Response APDU is almost the same as specified in the standard [3.2.2, 4.1.1.4, and table A5], with a new type of Database/Diagnostic-Record. The elements used in Database-Records [3.2.2.1.5, A.1.3.1] are specified in section 4 of this document.

The Result-Set will always be deleted by the DowQuest immediately after sending a Search-Response APDU.

The default element set returned in each Database-Record by DowQuest in a Search-Response APDU is "Document-Header," defined in section 5 of this document.

For records that are beyond the Medium-Set-Present-Number in the Search APDU, DowQuest will return the "Document-Short-Header" element set. This will probably not happen in normal circumstances since DowQuest returns a maximum of 16 documents. The origin can request the Date/Score/Headline/etc. elements by requesting a Document-Headline element set in subsequent Search APDUs. [??? Perhaps we should use message-length or buffer sizes to control this, instead?]

See Appendix B.4 for an example Search-Response APDU.

4. Element Sets supported by DowQuest

The elements supported by a particular target are outside the Z39.50 standard [3.2.2.1.3]. DowQuest will support the following Element-Set-Names. These are used in Search and Search-Response APDUs. Element-Set-Names is an optional field in Search APDUs [Table 2, Table 3].

Elements marked with a "*" can only appear in a Search-Response APDU, since the information is deleted with the Result-Set, so is no longer available when requesting text, i.e. the text headline and code elements should only be used with Type-1 queries.

The second column notes whether an element is Required, Optional, or Conditional in a given APDU.

The elements and their tag values are defined in section 5 of this document.

4.3 Document-Header

A Search-Response APDU contains one variable element:

Seed-Words-Used	O	ASCII

The rest of this element set is returned by default for each Database-Record in a Search-Response APDU:

	System-Control-Number	R	ANY
	Version-Number	O	integer
	Score *	O	integer
	Best-Match *	O	integer
[???]	Lines	O	integer
	Document-Length	O	integer
	Source	O	ASCII
	Date	O	ASCII
	Title	C	ASCII
	Geographic-Name	O	ASCII

4.4 Document-Text

This element set may be returned for each Database-Record in a Search-Response APDU in response to a Type-1 query:

Document-ID	R	ANY
Version-Number	O	integer
Document-Text	R	ASCII

4.5 Document-Short-Header

This element set is returned in the Database-Record in a Search-Response APDU for documents that are beyond the Medium-Set-Present-Number:

Document-ID	R	ANY
Version-Number	O	integer
Score *	O	integer
Best-Match *	O	integer
Document-Length	R	integer

4.6 Document-Headline

This element set is returned in a Search-Response APDU when requested in a Type-1 Query in a Search APDU for documents that were previously returned with Document-Short-Header element sets because of size restrictions:

Document-ID	R	ANY
Version-Number	O	integer
Source	O	ASCII

```
            Date                        O        ASCII
            Headline                    R        ASCII
            Origin                      O        ASCII
```

4.7 Document-Long-Header

This element set may be optionally requested in a Search APDU to be
returned in a Search-Response APDU:

```
            Document-ID                 R        ANY
            Version-Number              O        integer
            Score *                     O        integer
            Best-Match *                O        integer
            Document-Length             R        integer
            Source                      O        ASCII
            Date                        O        ASCII
            Headline                    R        ASCII
            Origin                      O        ASCII
            Stock-Codes                 O        ASCII
            Company-Codes               O        ASCII
            Industry-Codes              O        ASCII
            [??? what about more general codes, e.g. author, pricing,
            copyright?]
```

4.8 Document-Codes

This element set is returned in a Search-Response APDU when requested
in a Search APDU:

```
            Document-ID                 R        ANY
            Version-Number              O        integer
            Stock-Codes                 O        ASCII
            Company-Codes               O        ASCII
            Industry-Codes              O        ASCII
```

6. Data Element Definitions

Begin-Date-Range is the latest date for finding documents in a query
where Date-Factor is DF_LATER or DF_SPECIFIED_RANGE. Dates are ASCII,
of the form yyyymmdd.

Best-Match is the approximate byte offset within a document of the
highest-scoring portion of the document.

Chunk-Code specifies the size of chunks used in document regions. The
default value is 1. In DowQUest two Chunk-Codes are supported:
DowQuest will provide Chunk-Code=3 (Paragraph-ID) for relevance
feedback Type-3 Queries in a Search APDU, and Chunk-Code=2 (Line) for
text retrieval Type-1 Queries in a Search APDU. Chunk-Code=1 (Byte)
is the most general case. With this chunk size, Chunk-Marker and
Chunk-ID-Length are not used. The origin may indicate ranges of a
document in bytes by setting Chunk-Code=1 and providing pairs of
byte-offsets in a relevance feedback Type-3 query. Otherwise, the
origin indicates chunk ranges by specifying Chunk-Start-ID and
Chunk-End-ID.

Chunk-End-ID -- see Chunk-Start-ID.

Chunk-ID-Length specifies how many bytes Chunk-IDs will be. In
DowQuest Chunk-ID-Length for paragraphs is 3 bytes. The contents of a
Chunk-ID is opaque to the origin system. The value is used unchanged
when specifying a chunk range in a relevance feedback Type-3 query.

Chunk-Marker specifies an ASCII byte sequence that will occur in the
document text as a delimiter for the start of a chunk (except
Chunk-Code=1 (Byte) which has no markers). In DowQuest Chunk-IDs for
paragraphs are preceded by "<ESC>1" which is a two-byte Chunk-Marker.

Chunk-Start-ID and Chunk-End-ID are either Chunk-IDs (type ANY) that
were each marked with a Chunk-Marker in the text of a document
returned in a Search-Response APDU; or, if Chunk-Code=1, they are
integers containing byte offsets in the text of the document. They
delimit the beginning and end of a user-selected relevant region of
the document to be used for a relevance feedback query.

Company-Codes contains ASCII codes describing companies that are
mentioned in a document.

Date is the ascii date a document was published (yyyymmdd).

Date-Factor is one of: 1 "DF_INDEPENDENT", 2 "DF_LATER", 3
"DF_EARLIER", or 4 "DF_SPECIFIED_RANGE". The default is
Date-Factor=1, which specifies no special weighting of dates. The
other 3 values specify bonus scoring for documents with dates greater,
less than, or between specified dates, respectively. Date-Factor=2
uses Begin-Date-Range, Date-Factor=3 uses End-Date-Range, and
Date-Factor=4 uses both.

De-Highlight-Marker -- see Highlight-Marker.

Document-ID is a field that was previously returned in a
Search-Response APDU. It is unique in the database being searched.
It must be used in a Search APDU exactly as it was returned in a
Search-Response APDU. See Document-ID-Chunk.

Document-ID-Chunk is the same as a Document-ID element, except that it
must be followed by two or three chunk elements defining a fragment of
the document: Chunk-Code, Chunk-Start-ID, Chunk-End-ID. Chunk-Code is
optional; if Chunk-Code is missing, the previous value of Chunk-Code
in the current APDU is used; or if Chunk-Code never appeared in this
APDU, the default value is Chunk-Code=1 (Byte).

Document-Length is the length of the entire document in bytes.

Document-Text is a portion of a document text.

End-Date-Range is the earliest date for finding documents in a query
where Date-Factor is DF_EARLIER or DF_SPECIFIED_RANGE. Dates are ASCII,

of the form yyyymmdd.

Headline is a short ASCII description of the document for presentation
to the user. In DowQuest it is a maximum of 160 bytes [??? is this a
requirement?].

Highlight-Marker and De-Highlight-Marker are character sequences that
precede and follow text that may be displayed with highlighting. In
DowQuest, every searchable term is preceded by "<DC1>" (0x11) and
followed by "<DC3>" (0x13).

Industry-Codes contains ASCII codes describing industries that are
mentioned in a document.

Max-Documents-Retrieved is the maximum number of documents requested
by the origin in a Search APDU to be returned in a Search-Response
APDU. In DowQuest the default value is 16 [??? probably should not
have a default value?]. The target may return less than
Max-Documents-Retrieved documents.

Newline-Characters indicates what characters are used at the end of
lines. In DowQuest this is "<CR>" (0x0D).

Origin-City is an ASCII name of the city and/or country where a
document originated.

Present-Chunk-Code-Bitmap is a bitmap indicating what Chunk-Codes may
be used in a Present APDU to specify a text range of a document to be
returned. See Search-Chunk-Code-Bitmap for its definition. [??? This
is obsolete. Chunk-Codes must be worked out more completely.]

Score is a measure of how well the document matched the query. It may
be any integer value. [??? We may need to define a valid score range
to be used by all targets, or add a field in the Init-Response APDU to
specify the range for the current target.]

Search-Chunk-Code-Bitmap is a bitmap indicating what Chunk-Codes may
be used in a Search APDU query to specify a range of a document. The
bitmap indicates with a "1" in a bit position that the corresponding
code number will be accepted by the target system. For example, to
indicate that the target accepts Chunk-Codes 1 and 3 in a
Search APDU it would return Search-Chunk-Code-Bitmap with bits 1 and
three set to 1 and all other bits 0.

Seed-Words is a text string containing the initial seed words in a
relevance feedback Type-3 query.

Seed-Words-Used is the same format as Seed-Words except it contains
only words that actually matched some documents in the database. This
allows the user interface to give the user feedback about which seed
words were effective in a query.

Source is an ASCII string identifying the original source of a
document (e.g. newspaper name, journal title, etc.)

Stock-Codes contains ASCII stock ticker codes for companies that are
mentioned in a document.

Text-List is a list of text strings that are provided by the user.
They are document fragments that come from outside the DowQuest
database which the user wants to use in a search. They are processed
in the same manner as seed words except they are not given seed word
weight bonuses. **This would be a new feature of a query within
DowQuest, and would require changes to the Query Server as well as the
User Server portion of DowQuest. It will not be implemented for the
June '90 prototype.

User-Information-Length is the length of the entire user information
part of an APDU when it consists of more than one element.
User-Information-Length does not include itself in the length.

Version-Number is used to validate a local copy of a document's text.
If a document is modified in the target server, its Version-Number

must be incremented. If a document may not be cached, Version-Number
is set to 0. The default value is 0.

5.1 Tag Values of the Data Element

This table is an extension to the table 19 in section 4.1.3 of the standard.

Element	Tag	PDU	R/O/C
User-Information-Length[???]	99	Init-Response	C
		Search	C
		Search-Response	C
Chunk-Code	100	Search	O
Chunk-ID-Length	101	Init-Response	C
Chunk-Marker	102	Init-Response	C
Highlight-Marker	103	Init-Response	O
De-Highlight-Marker	104	Init-Response	C
Newline-Characters	105	Init-Response	O
Seed-Words	106	Search	C
Document-ID-Chunk	107	Search	O
Chunk-Start-ID	108	Search	O
Chunk-End-ID	109	Search	C
Text-List	110	Search	O
Date-Factor	111	Search	O
Begin-Date-Range	112	Search	O
End-Date-Range	113	Search	C
Max-Documents-Retrieved	114	Search	R
Seed-Words-Used	115	Search-Response	O
Document-ID	116	Search	O
		Search-Response	R
Version-Number	117	Search-Response	O
Score	118	Search-Response	O
Best-Match	119	Search-Response	O
Document-Length	120	Search-Response	R
Source	121	Search-Response	O
Date	122	Search-Response	O
Headline	123	Search-Response	C
Origin-City	124	Search-Response	O
Search-Chunk-Code-Bitmap	125	Search	O
Present-Chunk-Code-Bitmap [???]	126	Search	O
Document-Text	127	Search-Response	R
Stock-Codes	128	Search-Response	O
Company-Codes	129	Search-Response	O
Industry-Codes	130	Search-Response	O

Appendix A. Type-3 Query (Relevance Feedback)

Query syntax is not part of the Z39.50 specification, but a Type-1 query is suggested in Appendix B of the standard for Boolean queries. This is a similar suggestion for relevance feedback queries.

The Type-3 Query supports the relevance feedback style of database query (as provided by DowQuest). The Type-3 query includes the following elements:

Seed-Words	R	ASCII	
Document-ID	O	ANY	(see Note 1 below)
Document-ID-Chunk	O	ANY	(see Note 2 below)
Chunk-Code	O	binary	
Chunk-Start-ID	C	if Chunk-Code=1, binary else ANY	
Chunk-End-ID	C	if Chunk-Code=1, binary else ANY	

(may repeat Document-ID and Document-ID-Chunk elements)

Text-List	O	ASCII	(Not in DowQuest)
Date-Factor	O	integer	
Begin-Date-Range	C	ASCII	
End-Date-Range	C	ASCII	
Max-Documents-Retrieved	R	integer	

Note 1: There may be any number of Document-ID and Document-ID-Chunk elements in a Type-3 Query, intermixed.

Note 2: Each occurrence of a Document-ID-Chunk element must be followed by two or three chunk elements, defining a fragment of the document.

Appendix B. Sample APDUs in WAIS Demonstration System

In the following, binary values are shown in hexadecimal preceded by
0x. Variable fields include a tag and length [see A.1.2.1, A.1.2.2,
and Table 19]. See section 5.1 of this document for tag values for
WAIS elements.

B.1 Init APDU

[see Table 7, Table A2]

ITEM	BYTE POS.	VALUE	NOTE
Header-Length-Indicator	1-2	0x0015	21
Header:			
Fixed portion:			
PDU-Type	3	0x14	20
Variable Portion:			
Protocol-Version	4-6	0x030101	1
Options	7-9	0x0401C0	bit 1,2
Preferred-Message-Size	10-13	0x05020400	1024
Maximum-Record-Size	14-17	0x06020800	2048
Reference-ID	18-23	0x020400000001	1
User information part:			
(none)			

B.2 Init-Response APDU

[see Table 8, Table A3]

ITEM	BYTE POSITION.	VALUE	NOTE
Header-Length-Indicator	1-2	0x0025	37
Header:			
Fixed portion:			
PDU-Type	3	0x15	21
Result	4	0x01	1="accept"
Variable Portion:			
Protocol-Version	5-7	0x030101	1
Options	8-10	0x0401C0	bit 1,2
Preferred-Message-Size	11-14	0x05020400	1024
Maximum-Record-Size	15-18	0x06020400	1024
Implementation-Name	19-28	0x0908"DowQuest"	
Implementation-Version	29-33	0x1003"1.0"	
Reference-ID	34-39	0x020400000001	1
User-Information-Field	40-42	0x??0217	??
Search-Chunk-Code-Bitmap	43-45	0x7D0140	bit 2
Present-Chunk-Code-Bitmap??	46-48	0x7E0180	bit 1
Chunk-Id-Length	49-51	0x650103	3
Chunk-Marker	52-55	0x66021B6C	"<ESC>1"
Highlight-Marker	56-58	0x670111	"<DC1>"
De-Highlight-Marker	59-61	0x680112	"<DC2>"
Newline-Characters	62-65	0x69020D0A	"<CR><LF>"

B.3 Search APDU

[see Table 9, Table A4]

B.3.1 Example query containing only Seed-Words element (no
    Document-ID):

ITEM	BYTE POSITION.	VALUE	NOTE
Header-Length-Indicator	1-2	0x0018	24
Header:			
Fixed portion:			
PDU-Type	3	0x16	22
Small-Set-Upper-Bound	4-6	0x000400	1024
Large-Set-Lower-Bound	7-9	0x000800	2048
Medium-Set-Present-Number	10-12	0x000800	2048

ITEM	BYTE POSITION.	VALUE	NOTE
Replace-Indicator	13	0x01	1="on"
Variable Portion:			
Result-Set-Name	14-15	0x1100	" "
Database-Names	16-17	0x1200	" "
Query-Type	18-20	0x130133	"3"
Reference-ID	21-26	0x020400000002	2
User-Information-Field	27-29	0x??0224	36
Type-3 Query:			
Seed-Words	30-62	0x6A1F"Tell me about Thinking Machines"	
Max-Documents-Retrieved	63-65	0x720110	16

[??? remove this field; use Small-Set-Upper-Bound or something...]

B.3.2 Example query containing Seed-Words, one Document-ID and
      one Document-ID-Chunk element. This query includes seed word
      "Apple," and specifies using all of document 00000001WJ in the
      search, and paragraphs with IDS 005 through 007 from document
      00000023WJ:

ITEM	BYTE POSITION.	VALUE	NOTE
Header-Length-Indicator	1-2	0x0018	24
Header:			
Fixed portion:			
PDU-Type	3	0x16	22
Small-Set-Upper-Bound	4-6	0x000400	1024
Large-Set-Lower-Bound	7-9	0x000800	2048
Medium-Set-Present-Number	10-12	0x000800	2048
Replace-Indicator	13	0x01	1="on"
Variable Portion:			
Result-Set-Name	14-15	0x1100	" "
Database-Names	16-17	0x1200	" "
Query-Type	18-20	0x130133	"3"
Reference-ID	21-26	0x020400000003	3
User-Information-Field	27-29	0x??0230	48
Type-3 Query:			
Seed-Words	30-36	0x6A05"Apple"	
Max-Documents-Retrieved	37-39	0x720110	16

[??? remove this field; use Small-Set-Upper-Bound or something...]

Document-ID	40-51	0x740A00000001WJ	
Document-ID-Chunk	52-63	0x740A00000023WJ	
Chunk-Code	64-66	0x640102	paragraph
Chunk-Start-ID	68-72	0x6C03"005"	par ID=005
Chunk-End-ID	73-77	0x6D03"007"	par ID=007

B.4 Search-Response APDU

[see Table 10, Table A5]

ITEM	BYTE POSITION.	VALUE	NOTE
Header-Length-Indicator	1-2	0x0014	20
Header:			
Fixed portion:			
PDU-Type	3	0x17	23
Search-Status	4	0x00	0="success"
Result-Count	5-7	0x000002	2
Number-of-Records-Returned	8-10	0x000002	2
Next-Result-Set-Position	11-13	0x000000	0
Variable Portion:			
Present-Status	14-16	0x1B0100	0="success"
Reference-ID	17-22	0x020400000002	2
User-Information-Field	23-25	0x??01DD	221
Seed-Words-Used	26-44	0x7311"Thinking Machines"	
Database records:			
Document-Header element set:			
Document-ID	45-58	0x740C"0000000001WJ"	
Version-Number	59-61	0x750100	0
Score	62-67	0x760400000022	34
Best-Match	68-77	0x7708000000000000001	
Document-Length	78-87	0x7808000000000000033	

```
Source            88-92      0x7903"WSJ"
Date              93-100     0x7A06"900601"   yymmdd *
Headline          101-109    0x7B11"TMC Releases WAIS"
Origin-City       110-124    0x7C0D"Cambridge, MA"

Document-ID       125-138    0x740C"0000000123ZF"
Version-Number    139-141    0x750100              0
Score             142-147    0x760400000015   21
Best-Match        148-157    0x7708000000000000006E
Document-Length   158-167    0x7808000000000000000121
Source            168-182    0x790D"Business Week"
Date              183-190    0x7A06"900603"
Headline          191-211    0x7B13"Apple Releases WAIS"
Origin-City       212-226    0x7C0D"Cupertino, CA"
```

(*) A Date element should actually be yyyymmdd

Appendix C. DowQuest Code Formats

C.1 Company Codes

[??? TBD]

C.2 Industry Codes

[??? TBD]

C.3 Stock Codes

[??? TBD]